SH79F082 应用指南

1 概述

SH79F082 是一颗采用单机器周期增强型 8051 内核的单片机，内部采用 FLASH 存储程序代码，可以多次烧写或者擦除，方便用户调试和在线更新。

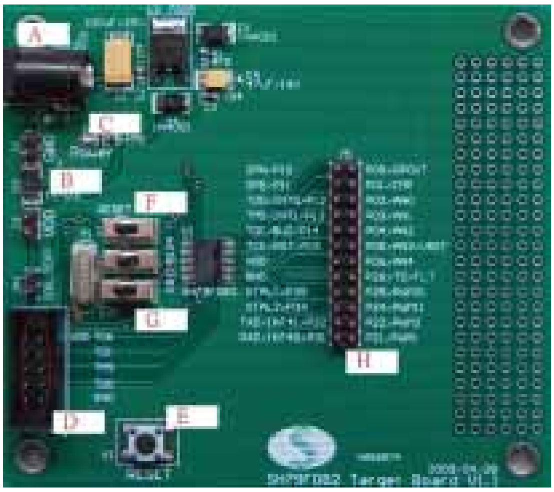
中颖 SH79F082 系列单片机的在线仿真和下载烧录可以通过 JET51 仿真器实现。该工具基于 Keil μ vision 集成开发环境，通过 JTAG 方式和单片机连接，实现全功能在线仿真和下载烧录功能。

1.1 封装

SH79F082 提供 TSSOP24 封装。

1.2 演示板（Target Board）

SH79F082 提供了一套演示板，供客户熟悉芯片功能。如下图：



SH79F082 演示板（TSSOP24）

板中各部分功能说明：

A. 外部供电接口 1 (内正外负)

外接 9~12V 电源，经过板上稳压电路产生 5V 电压作为芯片的供电电源使用。采用外部供电接口 1 供电，演示板供电电源为 5V，不可调。

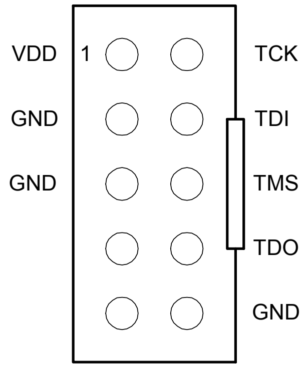
B. 外部供电接口 2

外部供电电源接口，分别与芯片的 VDD 和 GND 管脚连接。采用外部供电接口 2 供电，演示板供电电源可调。但注意供电电源不要超过芯片最高工作电压。

C. 电源指示灯

D. JTAG 仿真接口

JTAG 仿真接口和 JET51 仿真器一一对应，可用 10 芯扁平线直接连接。仿真时，演示板的供电电源可通过仿真接口的 VDD 管脚直接获得。



E. 复位按键

F. RESET 引脚拨位开关

板上可以选择 P1.5 用作 I/O 还是复位管脚 (通过下载时选择的“Option”来定)。若选择作为 I/O 使用，则将拨位开关拨到 IO 标识一侧（即靠近芯片一侧）。若选择作为复位管脚使用，则将拨位开关拨到 RESET 标识一侧（即靠近振荡器插座一侧）。

G. 振荡器引脚拨位开关

板上可以选择内部 RC 或外部晶体(16MHz)振荡器(通过下载时选择的“Option”来定)。若选择内部 RC，则 P3.4/P3.5 可用作 I/O 口，此时可将拨位开关拨到 IO 标识一侧（即靠近芯片一侧）。若选择外部晶体(16MHz)振荡器，则将拨位开关拨到另外一侧（即靠近振荡器插座一侧）。

H. 芯片的全部输入输出口或其它共享的功能口

此演示板和 JET51 配合使用，即可仿真 SH79F082 的全部功能。演示板的供电可以由外部供电接口 1 或外部供电接口 2 或仿真接口提供，不管使用哪一种供电方式，最终外部供电接口 1 处的电压为芯片的供电电压。

2 仿真

2.1 仿真软件的安装

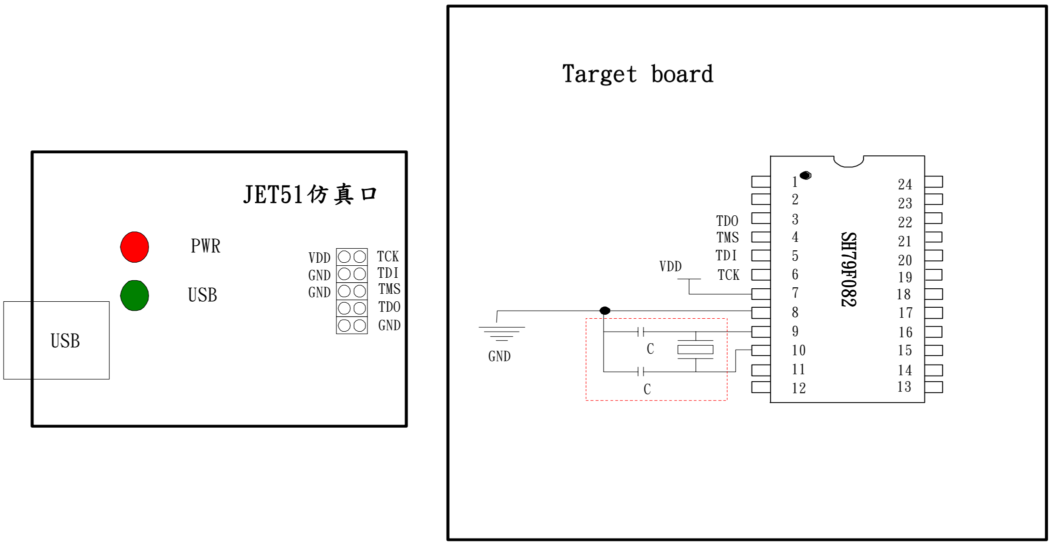
安装步骤：

1) 安装 Keil µ vision2（V2.34 及以上版本）或者 µ vision3。
2) 安装中颖 SH79F 组件库 JET51_Keil_Vxxx.exe（其中 Vxxx 表示版本号）到 Keil c 的安装目录下（例如: c:\keil\）。
3) 将 JET51 仿真器通过 USB 连接计算机，此时红色电源灯和绿色 USB 灯会同时亮起，说明安装成功。

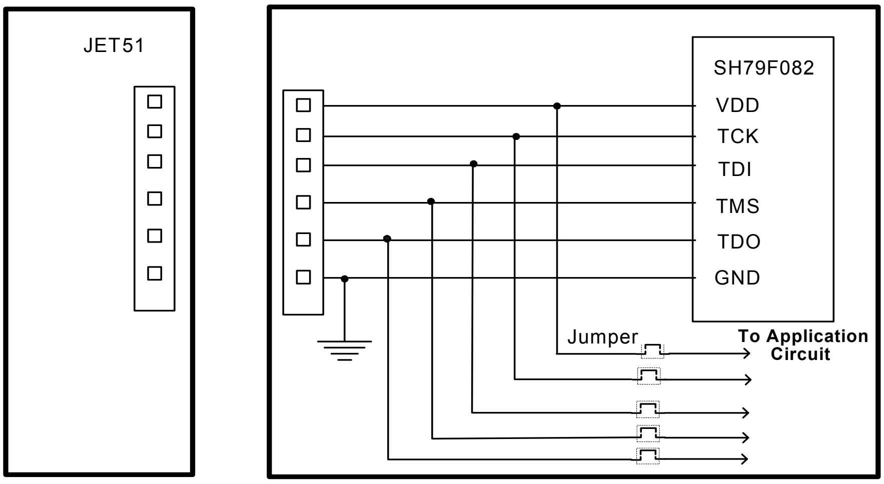
2.2 硬件的安装

安装步骤：

1) 将 SH79F082 用户目标板上的 JTAG 仿真口（VDD，TCK，TDI，TMS，TDO，GND）与 JET51 仿真器的 JTAG 仿真口通过扁平电缆相连即可。
2) 如果需要仿真，当使用晶振时，需要把振荡器引脚拨位开关拨到左侧；使用内部 RC 时，则把振荡器引脚拨位开关拨到右侧，振荡器输入输出引脚作为 IO 使用。
3) 当使用 JET51 仿真、下载时，JTAG 仿真口不能够用做 I/O，另外为确保正常通讯,最好将 TCK，TDI，TMS，TDO 与用户目标板上的外接线路断开,如果不断开,必须保证 TCK，TDI，TMS，TDO 负载电容不能超过 103（10000pF），否则通迅可能会失败。
4) 用户目标板电源上的负载电容不得超过 220µF，否则可能导致通迅失败。



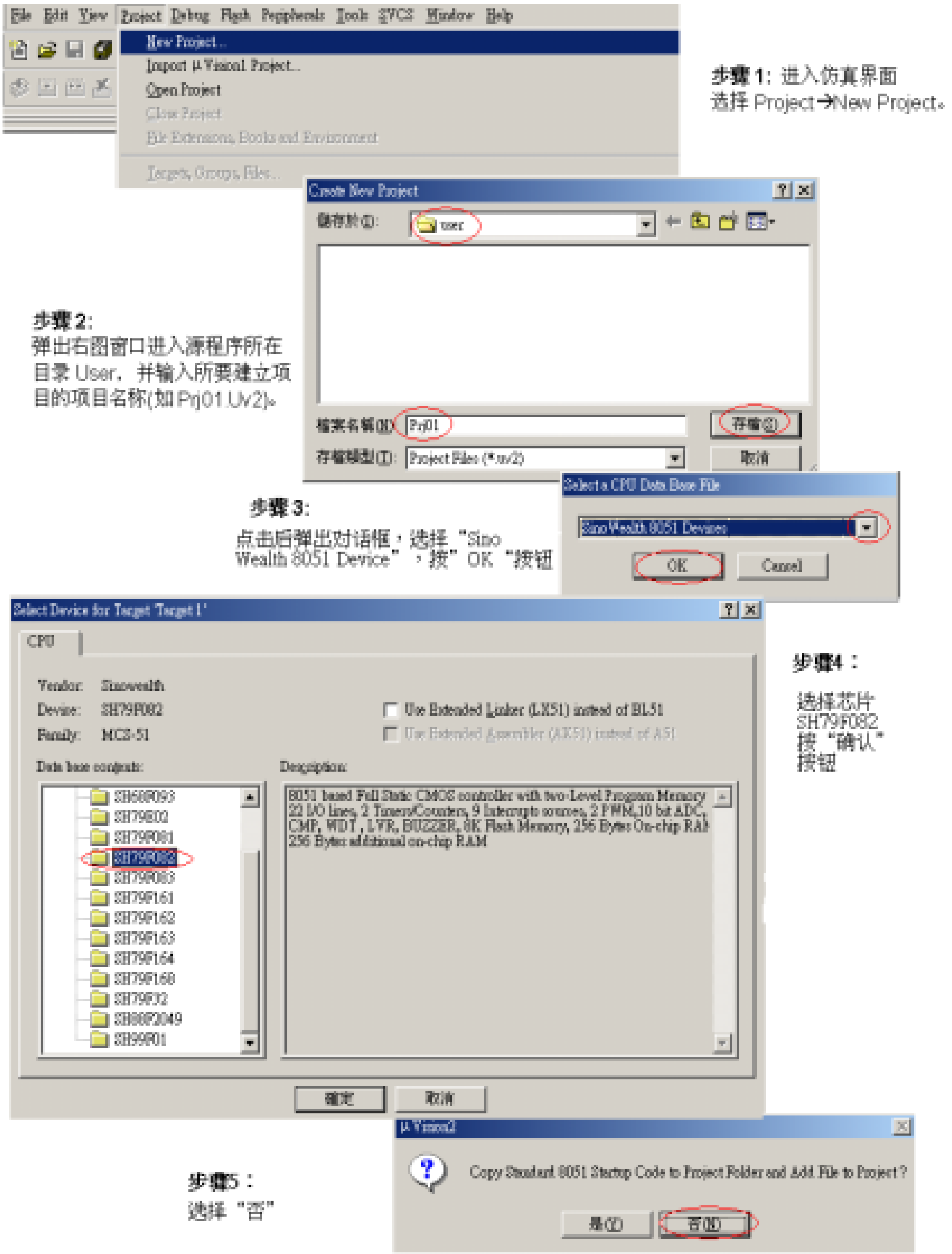
设计用户目标板时，最好将 JTAG 仿真口通过跳线与应用电路连接，以便烧写或者仿真时可以断开跳线与仿真器连接。



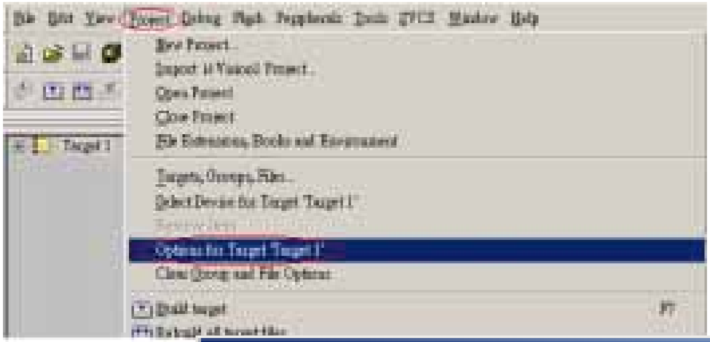
2.1.1 新建一个 Project

示例：用户的源程序在目录 User 下，文件名为 aa.asm。

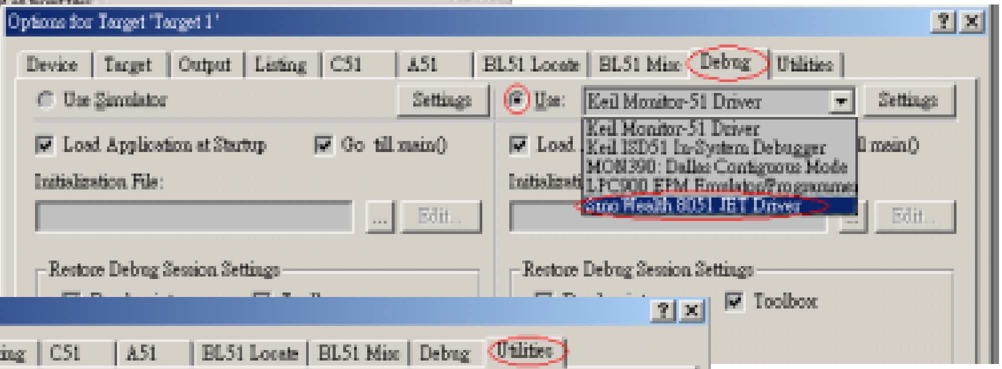
现在需要对此建立一个新的 Project，操作如下：



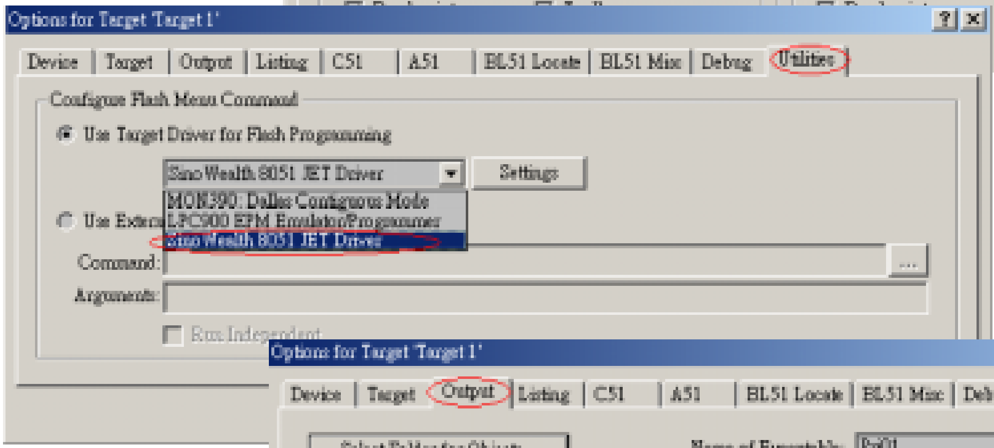
至此，已完成建立一个空的项目。下面设置项目属性。



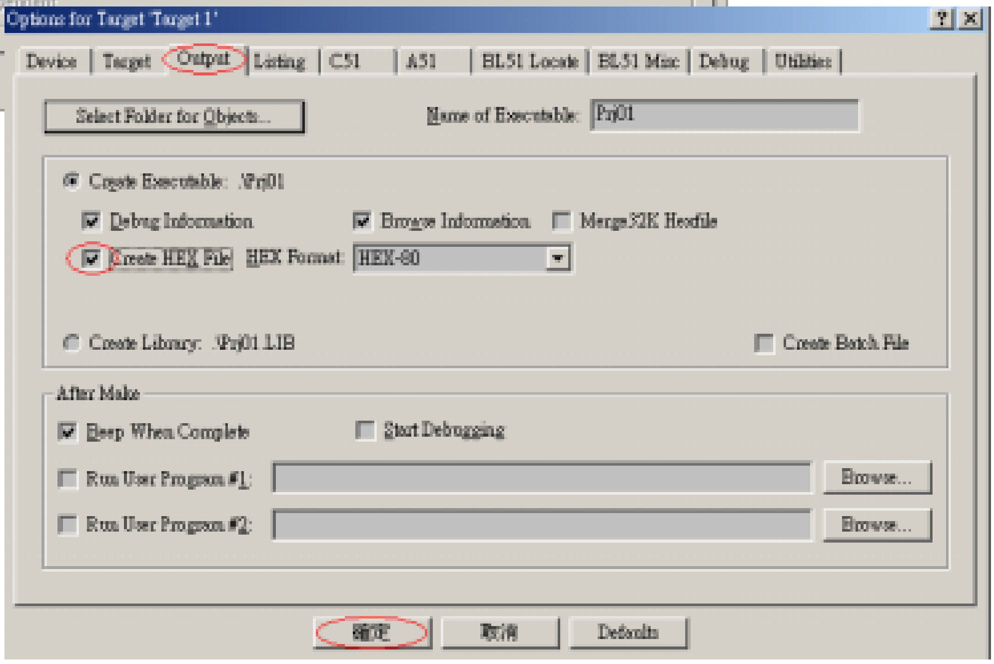
步骤 6:
选择 Project
→Options for Target ‘Target1’。



步骤 7:
选择 Debug→Use
→Sino Wealth 8051
JET Driver。

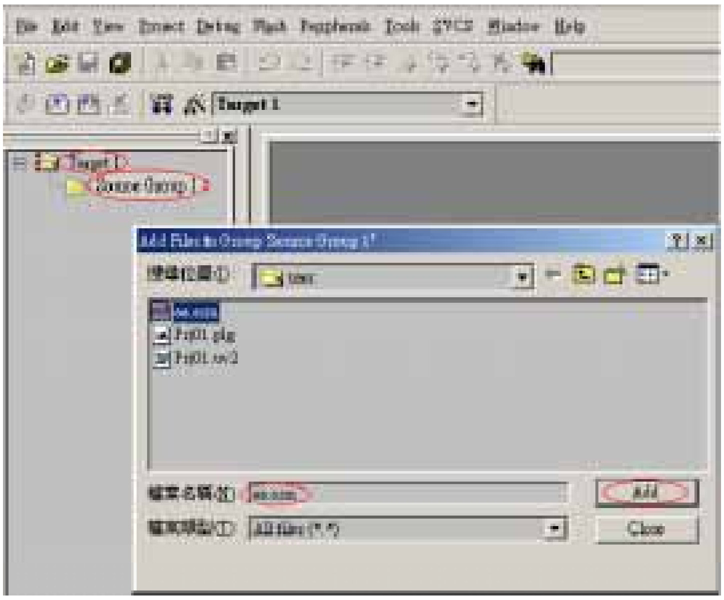


步骤 8:
选择 Utilities
→Sino Wealth 8051
JET Driver
→确认。



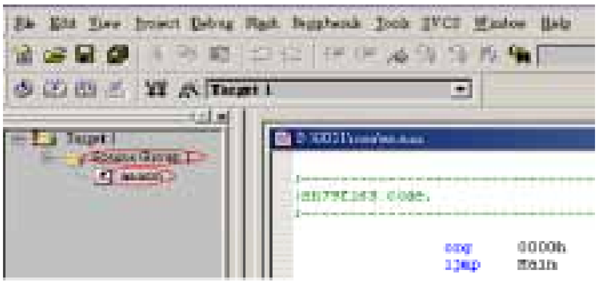
步骤 9:
选择 Output
→选择”Create HEX
File”→确认。

至此，项目属性设置完毕。下面将已经编写好的程序添加到项目中。



步骤 10:
双击 Target1
→选择“Source Group1”， 按右键
→选择目标目录，例如“User”
→选择目标文件，例如”aa.asm”
→点击“Add” 按钮。

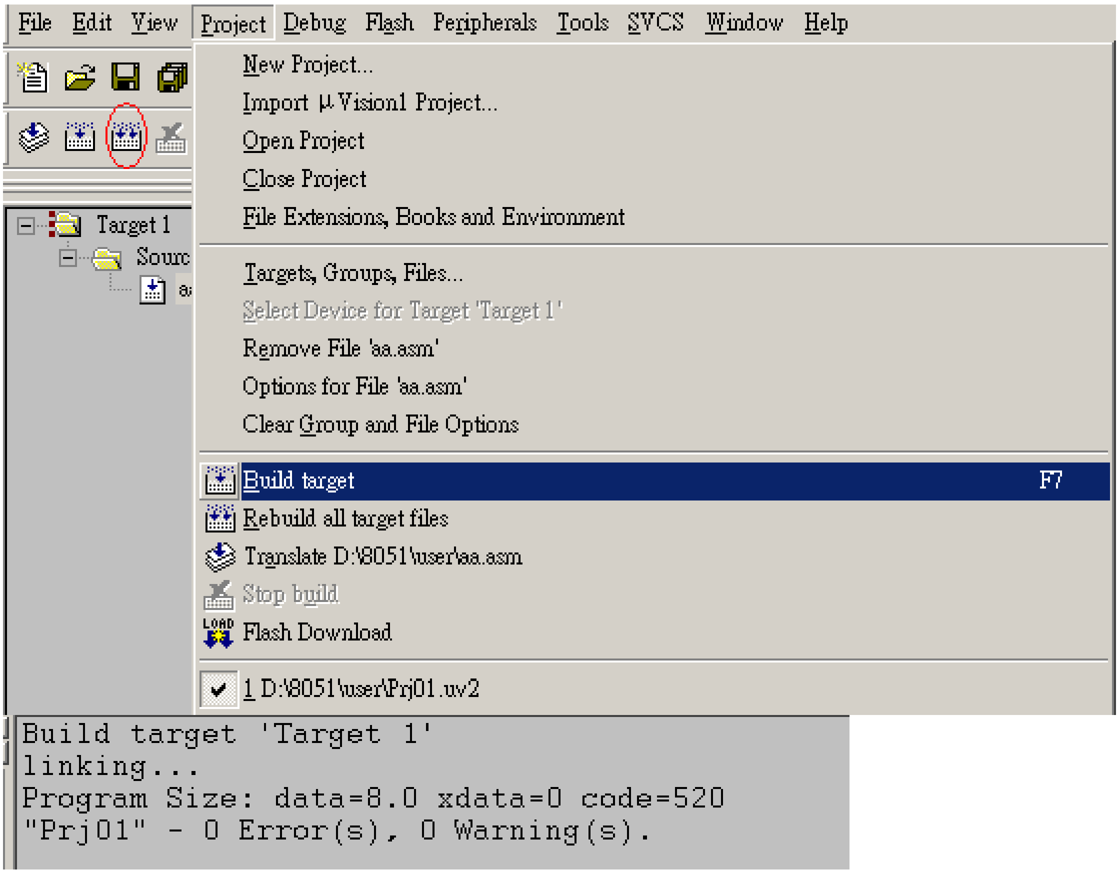
步骤 11:
关闭 Step10 中“Add Files to Group ‘Source Group1’”窗口
→双击“Source Group1”
→双击“aa.asm”。



2.1.2 程序的编译和下载

2.2.4.1 程序的编译

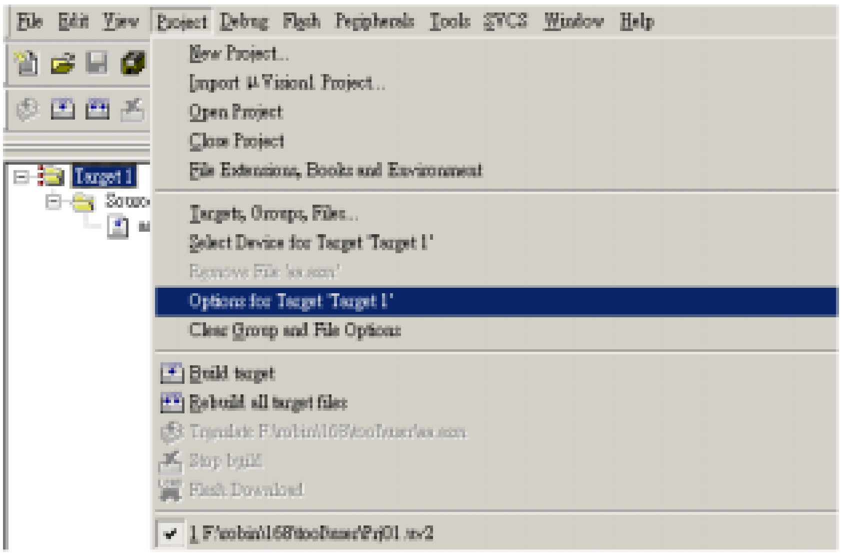
当程序编写完成后，点击菜单 Project\Build target(或编译按钮)，完成编译，生成.OBJ 文件和.HEX 文件。当存在编译错误(Error)，则不会生成.OBJ 文件和.HEX 文件。

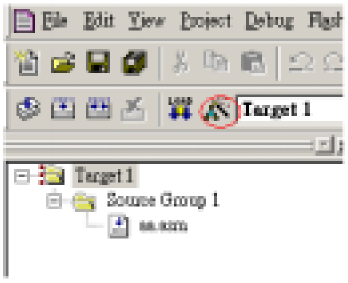


2.2.4.2 程序的下载

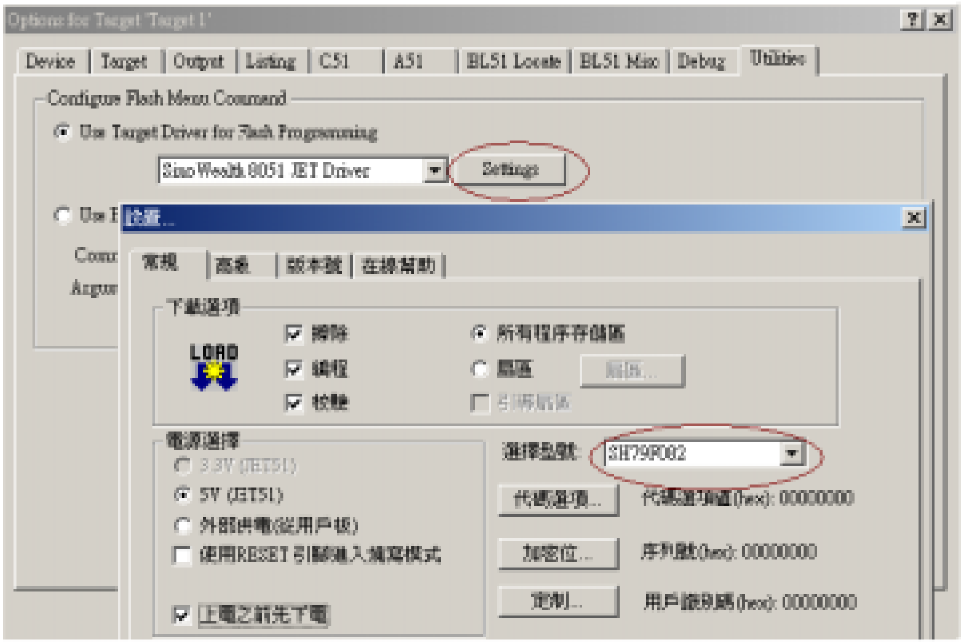
2.2.4.2.1 下载前的选项设置

程序下载前的选项设置包括选用芯片设置，上电方式设置，CODE OPTION 设置等。所有设置系统会自动保存。之后再进行程序下载时，可以跳过此步骤。

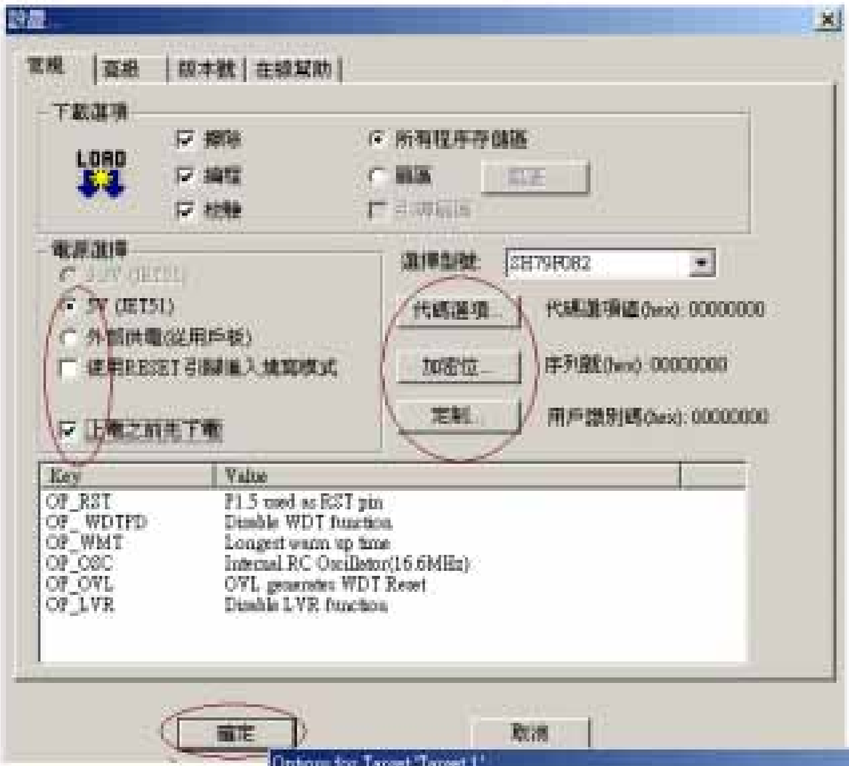




步骤 1:
点击 Project
→Option for Target 'Target 1'
(或直接点击快捷按钮)。



步骤 2:
弹出右图窗口后点击
Utilities
→Setting
→选择
"SH79F082"



步骤3：

用户根据需要选择上电方式及option等选项，全部选择完成后点击"确定"按钮

步骤4:

按"确认"按钮完成设置.

注意 JET51 仿真器提供了 2 种不同的上电方式，在步骤 3 中设置。

1) 方式 1: JET51 仿真器供电

选择“5V(JET51)”和“Power Off，Power On again”。其中，“Power Off，Power On again”是指“下电/上电后重进 JTAG 模式”， 通常在仿真或者下载出错时选用此方式。

在程序下载和仿真的时候，用户目标板的供电由 JET51 仿真器提供。此时，用户目标板不需要外接任何电源，电源会通过 JET51 仿真器电源直接供给用户目标板。

如若用户目标板耗电小于 20mA，推荐用方式 1。

2) 方式 2: 外部供电

选择“External(Target board)”。

在程序下载和仿真的时候，用户目标板通过外接电源供电。此时，上电顺序有严格要求。

步骤 1：确认用户目标板处于断电状态。

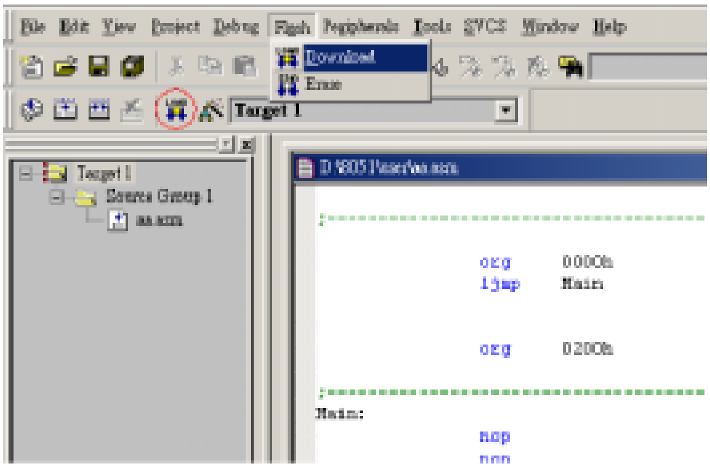
步骤 2：连接 JET51 仿真器和用户目标板。

步骤 3：JET51 仿真器和 PC 通过 USB 连接，红色电源灯和绿色 USB 灯全部点亮。

步骤 4：用户目标板接入外部电源。

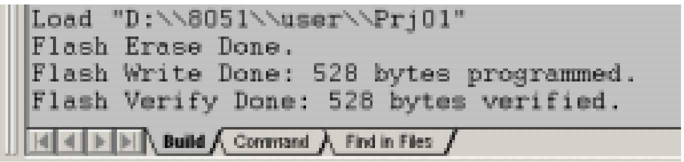
此后调试可以反复下载和仿真，无须下电及重新连接。

2.2.4.2.2 程序下载



按 “Download” 快捷按钮
或点击 Flash→Download。

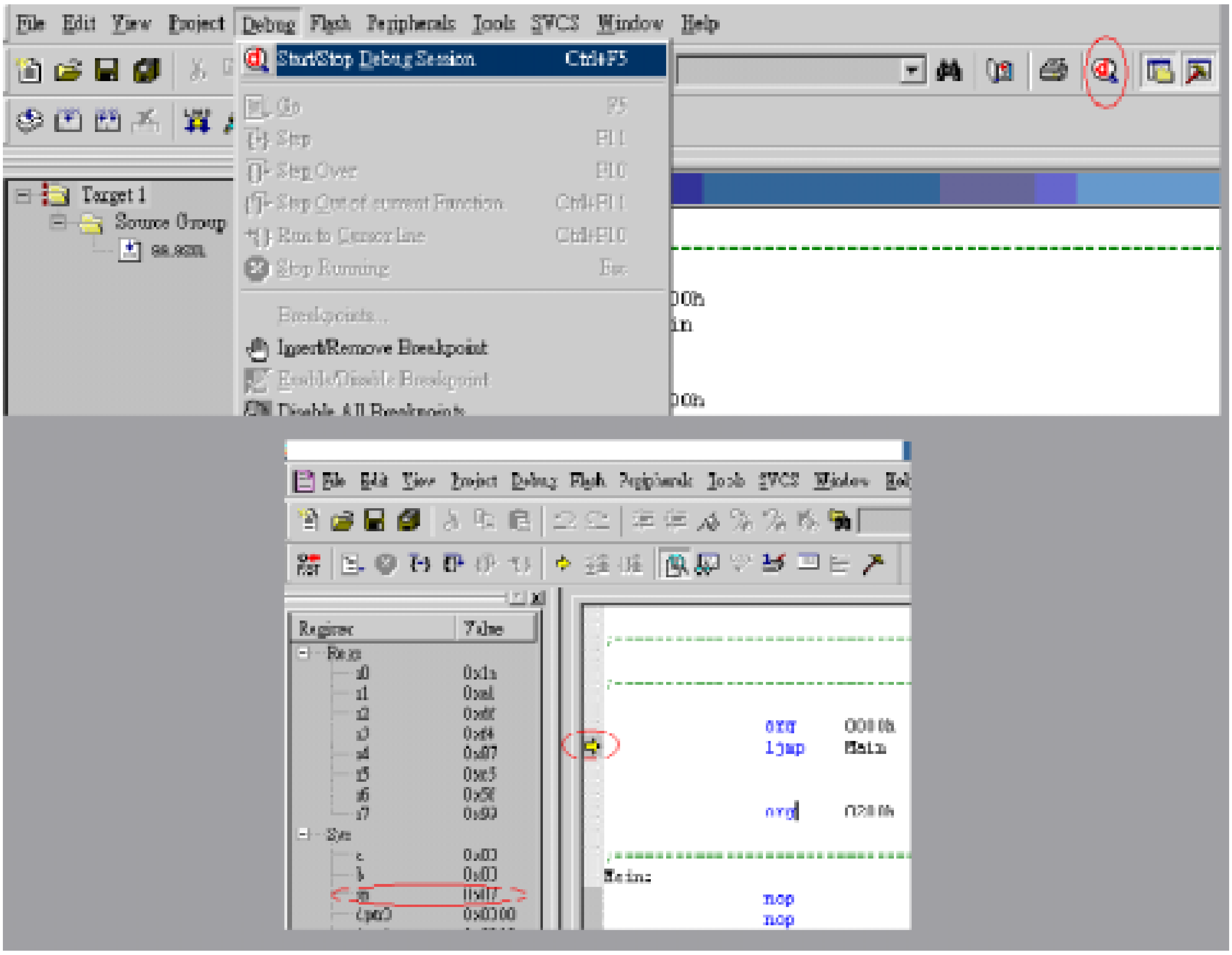
Download成功。



2.1.3 程序仿真

当程序已经下载到芯片中，点击菜单 Debug\Start/Stop Debug Session 或快捷按钮进入程序仿真模式。

当 JTAG 完全连接上以后，会有黄色箭头指向 0000H 处，同时左边寄存器窗口 SP 的值为 07H，表示成功进入调试模式可以进行仿真了。用户可以通过选择 Debug 菜单下的命令来执行 Step、Step Over、Run、Stop 等操作，也可以执行增加断点、减少断点的操作。



注意:

1) 断点为将该指令执行完以后，再停止。
2) 执行 Step Over 指令，在执行到 ACALL、LCALL 等语句时，会将 ACALL、LCALL 下一条语句执行完才停止。
3) 仿真器只支持 7 个断点。若超过 7 个断点，则只支持最后设置的 7 个断点。以前的断点将被忽略。
4) 程序仿真无误后，断开外加电源，拔掉 JTAG 连线。
5) 用户目标板接入外部电源，开始独立运行（脱机模式）。

3 应用注意事项

3.1 和传统 8051 差异

SH79F082 是一颗采用增强型 8051 内核的单片机，功能比传统型 51 内核有以下增强；

3.1.1 SH79F082 机器周期为 1 个振荡器周期，而传统型 51 核机器周期为 12 个振荡器周期。其指令执行效率约为同频率的传统 8051 的 10 倍。详细指令执行时间参见 SPEC。

3.1.2 SH79F082 系统时钟可以 1,2,4,12 分频,上电后默认为 12 分频。

3.1.3 SH79F082 提供增强型的乘/除法指令，支持 16BIT × 8BIT 和 16BIT ÷ 8BIT 运算。

3.1.4 SH79F082 提供双 DPTR 指针，用户在使用查表，数据搬移等操作时会更方便。

3.1.5 SH79F082 复位脚采用低电平复位，不同于传统 8051 的高电平复位。

其它功能上的差异请参见 SPEC。

3.2 FLASH ROM

3.2.1 SH79F082 每 1K BYTE 为 1 个扇区（SECTOR），除了最后一个扇区因为存放有 OPTION 信息被硬件设定为不能做扇区擦除(SECTOR ERASE)外，其它每个扇区可分别进行擦除。全局擦除(MASS ERASE)可以擦除所有扇区。

3.2.2 SH79F082 可以对 flash 进行加密，加密以 2K 为一个单位。

3.2.3 最后一个扇区的最后 64BYTE（SH79F082 ROM 地址的 1FC0H~1FFFH）保留，不可以用作程序 ROM（硬件屏蔽）。

3.2.4 存放在 FLASH ROM 中的程序可以对不在同一扇区的 FLASH ROM 编程（SSP 模式），因此可以当作 EEPROM 来使用。扇区中的数据从 0 改为 1 时，可以直接烧写。但从 1 改为 0 时，必需通过扇区擦除后才能实现。

3.2.5 在进行 FLASH 编程时，系统时钟不能低于 1M。

3.2.6 SH79F082 支持 ICP 功能，即在客户应用电路板完成后，在客户电路板上通过烧写器烧写程序代码。但须注意，P1.2~P1.5 四个烧写口不能有大于 10nF 的电容性负载，VDD 管脚上的负载电容不得超过 220μF，否则可能导致烧写失败。

3.2.7 FLASH 烧写/擦除步骤。

3.2.7.1 烧写

1) 关闭中断。在整个烧写期间无法响应中断，但中断申请标志位照常会置 1。因此如果烧写期间来了中断，烧写完成后，打开中断允许位，中断服务程序仍将执行。
2) 设置 XPAGE，IB_OFFSET 寄存器。此 2 个寄存器对应烧写地址。其中 IB_OFFSET 对应地址低位，XPAGE 对应地址高位。
3) 将准备填充值写入 IB_DATA。此寄存器中的值将被写入到 FLASH 中指定地址。
4) 按照顺序设置 IB_CON1~5。其中数值如下所列。

IB_CON1----→ #6EH ;表示烧写。

IB_CON2 ---→ #05H
IB_CON3----→ #0AH
IB_CON4 ----→ #09H
IB_CON5 ----→ #06H

注意顺序不能有错，数值也必须正确。烧写完成后这 5 个寄存器硬件自动清零，防止误写。因此下次写操作必须重新填充这 5 个寄存器。

5) 软件写入 IB_CON5 后启动硬件写操作，CPU 将进入 IDLE 模式。烧写完成后自动唤醒。此步骤由 CPU 自动完成，用户无须干预。
6) 软件设置完上述寄存器后插入 4 个 NOP 指令，保证烧写完成后 CPU 从 IDLE 模式可靠唤醒。
7) 如需继续写入数据，跳转至第 3 步。
8) 将 XPAGE 寄存器清除为零，恢复中断设置。
9) 程序示例：注意示例中各数据均为示意性数据，应用时请根据具体情况设置。

```
PUSH    IEN0
CLR     EA                  ;步骤 1
MOV     XPAGE,#18H          ;步骤 2
MOV     IB_OFFSET,#00H
MOV     IB_DATA,#05H        ;步骤 3
MOV     IB_CON1,#6EH        ;步骤 4
MOV     IB_CON2,#05H
MOV     IB_CON3,#0AH
MOV     IB_CON4,#09H
MOV     IB_CON5,#06H
;/*步骤 5: CPU 进行烧写，自动进入 IDLE 模式，完成后自动唤醒*/
NOP                         ;步骤 6
NOP
NOP
NOP
;/*步骤 7: 如果需要继续写入数据，重复上述步骤*/
MOV     XPAGE,#00H          ;步骤 8
POP     IEN0
```

3.2.7.2 擦除：

1) 关闭中断。在整个烧写期间无法响应中断，但中断申请标志位照常会置 1，因此如果擦除期间来了中断，擦除完成后，打开中断允许位，中断服务程序仍将执行。
2) 根据需要擦除的扇区设置 XPAGE。1K BYTE 为一个扇区， SH79F082 ROM 为 8K

BYTE，XPAGE　BIT7~2 对应扇区号, XPAGE　BIT1~0 对应被擦除扇区高两位，IB_OFFSET 寄存器无效。

3) 按照顺序设置 IB_CON1~5，其中数值如下所列。

    IB_CON1----→ #0E6H　　　;表示擦除
    IB_CON2 ---→ #05H
    IB_CON3----→ #0AH
    IB_CON4 ----→ #09H
    IB_CON5 ----→ #06H

   注意顺序不能有错，数值也必须正确。擦除完成后，这 5 个寄存器硬件自动清零，防止误擦除。因此下次擦除操作必须重新填充这 5 个寄存器。

4) 软件写入 IB_CON5 后启动硬件擦除操作，CPU 将进入 IDLE 模式。擦除完成后自动唤醒。此步骤由 CPU 自动完成，用户无须干预。
5) 软件设置完上述寄存器后插入 4 个 NOP 指令，保证擦除完成后 CPU 从 IDLE 模式可靠唤醒。
6) 如果需要继续擦除扇区，跳转至第 3 步。
7) 将 XPAGE 寄存器清除为零，恢复中断设置。
8) 程序示例：注意示例中各数据均为示意性数据，应用时请根据具体情况设置。

```
PUSH    IEN0
CLR     EA                      ;步骤 1
MOV     XPAGE,#18H              ;步骤 2
MOV     IB_OFFSET,#00H
MOV     IB_DATA,#05H            ;步骤 3
MOV     IB_CON1,#0E6H           ;步骤 4
MOV     IB_CON2,#05H
MOV     IB_CON3,#0AH
MOV     IB_CON4,#09H
MOV     IB_CON5,#06H
;/*步骤 5: CPU 进行烧写， 自动进入 IDLE 模式，完成后自动唤醒*/
NOP                             ;步骤 6
NOP
NOP
NOP
;/*步骤 7:  如果需要继续写入数据，重复上述步骤*/
MOV     XPAGE,#00H              ;步骤 8
POP     IEN0
```

3.3 RAM

3.3.1 SH79F082 有 256 BYTE 内部 RAM，256BYTE 外部 RAM。

3.3.2 内部 RAM 地址范围为 0~FFH，其中 0~7FH 既可以直接寻址，也可以间接寻址，80~FFH 只能用间接寻址。若用直接寻址访问此地址范围，访问到的是系统寄存器。

3.3.3 外部RAM地址范围为0~FFH。传统的8051用MOVX A，@Ri (i=0或1)能访问0~FFH地址范围内的外部 RAM。

3.4 中断

3.4.1 OVL 复位:

SH79F082 为进一步增强 CPU 运行可靠性，内建程序超范围溢出检测电路，一旦检测到程序计数器的值超出 ROM 最大值，或者发现指令操作码(不检测操作数)为 8051 指令集中不存在的 A5H，便认为程序跑飞，产生 CPU 复位信号，同时将 WDOF 标志位置 1。为应用这个特性，用户应该将未使用的 Flash ROM 用 0xA5 填满。

附: 在程序中用 A5H 填充未用到 ROM 地址的方法:

首先定义宏:

```
Fill_A5H    MACRO    N
REPT        N
DB          A5H
ENDM
ENDM
```

然后在没有用到的 ROM 空间调用宏，比如 1000H 到 11FFH 没有用到，可以用如下方法填充:

```
CSEG AT     1000H
Fill_A5H    512
```

3.4.2 PORT 口外部中断

1) 由于 I/O 复用功能比较多，只有当相应的 PORT 口设置为 I/O 功能而且设置为输入状态，才有可能产生外部中断。内部上拉功能也需要软件控制。注意，内部上拉电阻大概在 30KΩ左右，如果刚开始 PORT 口输出低电平，中断初使化程序中将 PORT 口设为输入并且将内部上拉功能打开，然后马上开启 PORT 口中断控制位，而且设置为上升沿触发，由于引脚上有电容负载，内部上拉比较慢，内部中断上升沿检测电路可能会检测到此上升信号，产生一次中断请求。因此，建议此情况下，内部上拉打开后，等待一段时间（比如 5 个 NOP 指令）后，清除中断标志位,再打开中断允许控制位。

2) 所有 PORT 口中断，如果设为电平触发，则 FLAG 标志位仅仅受外部 PORT 口电平控制，电平为低，则 FLAG 为 1，电平为高，则 FLAG 自动变为 0（和传统 8051 兼容）。但是要注意外部中断 4 有多个中断源，需要在中断服务程序中查询标志位来确认是哪

一个中断源触发中断，如果外部中断低电平持续时间很短，小于中断响应时间，则中断服务程序中会查不到中断标志位，因此外部中断 4 设置为电平触发时，需要保证低电平持续时间大于中断响应时间（微秒级）。设为边沿触发时无此问题。如果设置外部中断 4 电平触发模式来唤醒 POWER DOWN 模式，由于进入 POWER DOWN 模式后，唤醒需要 2ms 的 WARM UP 时间，因此如果外部中断 4 低电平持续时间小于 2ms，中断服务程序中会检测不到中断标志位，无法区分是哪一个中断源唤醒。

3.5 IDLE & POWER DOWN 模式

程序编写时注意 IDLE 或 POWER DOWN 指令之后务必加至少 3 条 NOP 指令，以保证系统能够稳定进入 IDLE 或者 POWER DOWN 模式。

```
……
MOV     SUSLO，  #55H
ORL     PCON，#01H;idle mode (or #02H: power-down mode)
NOP                 ;必须添加
NOP                 ;必须添加
NOP                 ;必须添加
……                  ;后续程序。调试程序时如果需要在 Idle 唤醒处设置断点，
                    ;断点位置要放在 3 个 NOP 之后。
```

3.6 PWM

3.6.1 SH79F082 中有 2 路独立的 PWM，每路 PWM 提供两路输出，输出波形可以选择中心对齐和边沿对齐模式。每路输出有单独的允许位，以及单独的高低电平控制位。

3.6.2 PWM 有 3 个中断源，分别为周期溢出中断（因为周期寄存器共享,因此只有一个周期中断），占空比溢出中断（2 路 PWM 输出各占一个中断）。他们共享一个中断入口地址，有各自的使能位。

3.6.3 PWM 模块具有 12BIT 周期以及 12BIT 占空比寄存器（各自的占空比分别可调）和 12BIT 死区控制寄存器，设置寄存器时，需要按照先写低位，后写高位的顺序。如果只写低位，内部寄存器值不会更新。

3.6.4 PWM 有效期间所有寄存器均可任意修改。修改后的内容会在下一个 PWM 周期生效，但 PWM 输出控制寄存器是修改后立即生效。

3.6.5 设置 PWM 寄存器前需要先将 PWMLO（$E7）寄存器值设为 55H，否则硬件拒绝修改寄存器。

3.6.6 PWM 模块具有故障检测功能，当检测到有效故障信号后,则 PWM 输出会立即变为无效电平。

3.7 ADC

3.7.1 SH79F082 内部集成 10BIT 分辨率 ADC，ADC 时钟可由寄存器 TADC[2:0]设置为系统时钟 2 分频，4 分频，6 分频，8 分频，12 分频，16 分频，24 分频，32 分频。不论寻用多少频率的系统时钟，都需要保证 ADC 时钟周期大于或等于 1μs。否则会影响精度。如果实际使用中只需要 8BIT 精度，ADC 时钟周期可以更短一些。

3.7.2 完成一次 A/D 转换需要的时间分 2 部分。首先是采样时间，可由寄存器 TS[3:0]设置，最低 2 个 ADC 时钟周期。然后是转换时间，固定为 12 个 ADC CLOCK，因此完成一次 A/D 转换最少 14μs。外部仿真输入信号必须保证在采样期间固定，而在转换期间输入信号发生变化不会影响到最终 A/D 转换结果。

3.7.3 ADC 内部集成了数字比较器功能，可以将数字比较器参考值写入数值寄存器，将此功能打开（ADCON=101X XXX1B），ADC 一直处在转换状态，但每次转换完后不会产生中断标志。如果发现转换结果比参考值大，则会产生中断标志。因为占用了内部 AD 资源，因此 ADC 功能和数字比较器功能无法同时使用。

3.8 振荡器

3.8.1 25℃下内部 RC 振荡器频率为 16.6MHz±1%。如果温度变化，频率会飘移（温度提高，RC 频率会降低），-40℃~+85℃漂移 16.6MHz±2%。

注意:上面说的频率是取 1024 个时钟周期取平均值.

3.9 看门狗

3.9.1 内置有看门狗电路，溢出时间从 1ms 到 4s 可调整，如果在设定时间内没有操作看门狗寄存器（读或写），看门狗电路会产生复位信号，同时将标志位置 1。

3.9.2 看门狗电路拥有单独的内部 RC 振荡器，但频率不精确，SPEC 上定义的是各种最坏情况下的最短溢出时间。

3.9.3 看门狗电路耗电约 1uA,它可以在代码选项里选择关闭或打开（在 power down 下有效或无效）。

3.10 UART

3.10.1 TXD/RXD 引脚都和其它功能复用。 如果 REN 控制位设为 1，则 RXD 引脚自动设为输入，内部自动拉高。而 TXD 引脚只有在 UART 发送即软件向 SBUF 寄存器写入数据时才会作为 TXD 引脚使用，发送完成后又恢复为 I/O 功能。为了避免接收端误判，在 UART 发送初始化程序中，应该将相应 I/O 状态设为输出高电平。

3.10.2 UART 波特率发生器可以采用 Timer1 或系统时钟，UART 接收数据的波特率最大允许偏差为: ±4%。

3.11 电流采样模块

3.11.1 电流采样模块集成比较器放大器，比较器集成数字滤波功能，比较器输出可直接关闭 PWM，放大器输出可直接作为 ADC 输入。

3.12 可读识别码

3.12.1 在 SH79F082 每颗芯片中都有一个 8 位的可读识别码，值为 0-255 的随机值，是无法擦除的。它可以由程序或编程工具读出，此功能可以供软件加密使用。

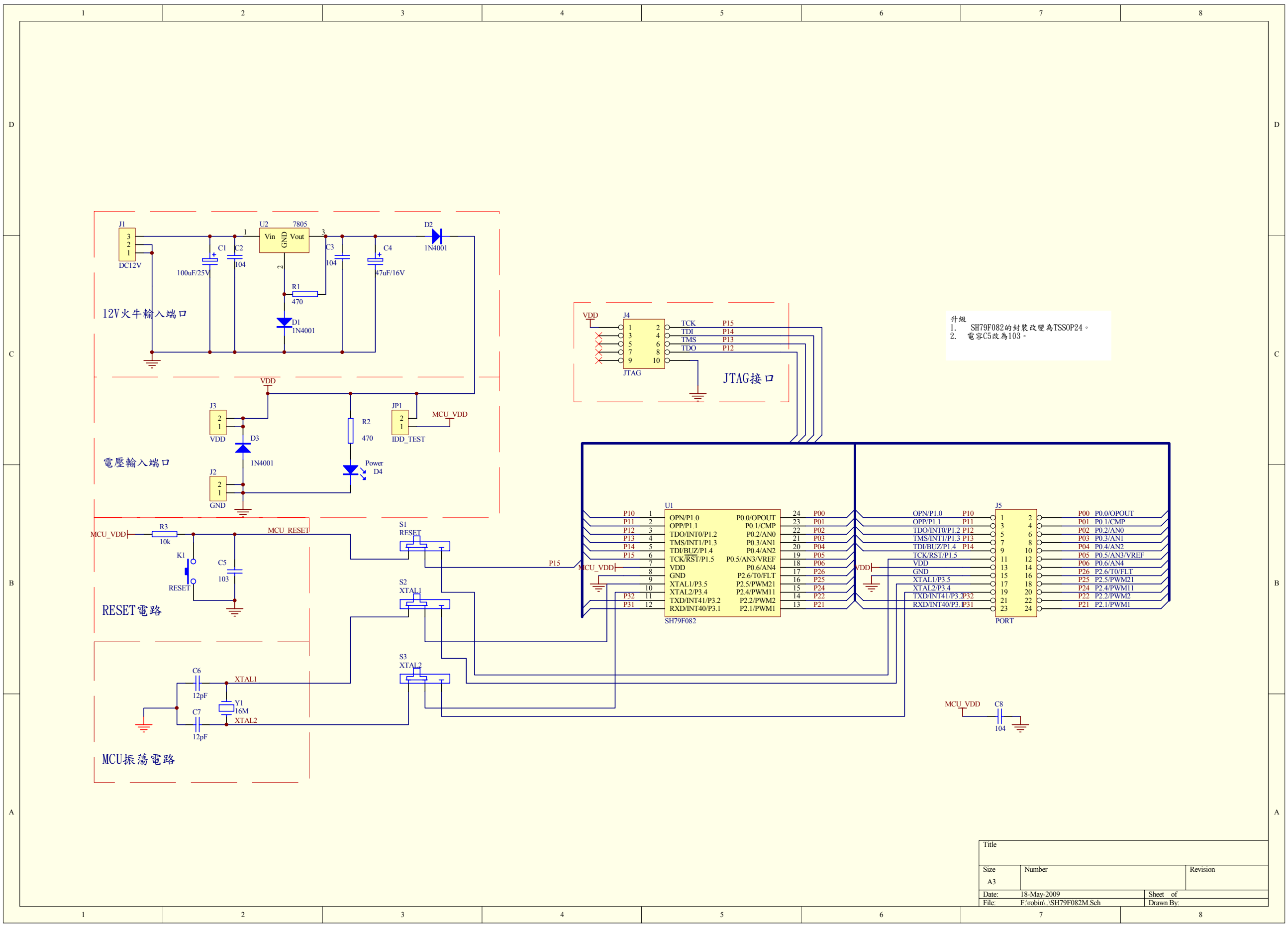
3.12.2 读识别码时，首先，设置 FAC 位为 1，然后给 DPTR 赋值 “007FH”，将 A 清 0，再使用 “MOVC A， @A+DPTR”来读取。

3.12.3 读完随即码后必须把 FAC 位清 0。

3.13 IO 口

3.13.1 在 spec 中没有用到的端口,应该设为输入上拉,否则可能会导致芯片内部漏电.漏电电流大概在几微安到几百微安范围内.

3.14 演示板原理图



3.15 PCB 布线注意

3.15.1 JTAG 接口（P1.2-P1.5）负载电容不能超过 103。特别是 TCK 和 reset pin 复用，其上电容不能超过 103。

4 应用指南版本历史

版本号	版本记录	时间
1.0	初始版本	2009.05